mods, fixes & upgrades

The Midwest Atari Group- Iowa Chapter presents

mods, fixes & upgrades

for 8-bit Atari Home Computers

Welcome to a world of change. Not that the 8-bit (or "dollar" type) Atari home computers aren't already the best computing value for the money, but most of us who own Atari's like to push our machines to the limit. And pushing means changing. Changing the amount of Random Access Memory (RAM). Changing the way different peripheral devices are handled. Changing the way Atari made the machine, to make it even better.

The articles in this booklet are all written by people just like you and me. Well, maybe a little more technically inclined, but they all own Atari's. These articles are all in the public domain, which means the authors aren't making any money for their labor. They wrote them for personal needs, to prove something was possible or just for the fun of it. And they are presented in that spirit. M.A.G.I.C. is not trying to make lots of money by putting this collection together. In fact, we're only trying to cover costs. But we felt these modifications, fixes and upgrades deserved as wide a distribution as possible, and this booklet is an attempt to attain that.

These articles are presented as is, without warranty. They have been done by the authors and probably some others, but there are no guarantees. In almost every case, doing these mods will result in the voiding of any warranty in force, so please consider that before you start in. It is assumed that these mods will be performed by someone who knows what they're doing. If you don't, stop right here and find a book or class that will teach you the rudiments of electronics. Basic rules of electronics safety should be followed. Anyone who injures themselves while doing one of these mods has only themselves to blame. Please be careful and use common sense.

Every effort has been made to avoid that demon of the publishing business, typographical errors. However, some may have slipped by. If you purchased this booklet by mail, we'll try to send out any corrections we find are necessary. And we'll try to let you know about any future projects.

With those warnings aside, we now open the door to that world of change. The world of enhanced Atari power. Have fun!

The Midwest Atari Group- Iowa Chapter -- August 1987

mods, fixes & upgrades
table of contents

The 800- Plus 256K 2

800 +++++ rek 256-512-1-2-4 MEG!! 8

Atari 800XE Modification 10

Souping Up the 1200XL 13

130XE/320K Upgrade 15

A word from our sponsor 17

130XE/576K Upgrade 18

XEMEG- The 1 Meg 130XE 20

Newell 256K Upgrade Fix 33

XE Audio Fix 33

800XL Keyboard Fix 35

XM301 Fix 36

XE Console Key Fix 37

Hardwired Keypad 39

4-Joystick XL/XE 41

1050 Write Enable Switch 45

Shared Drives 47

1050 Double Density Fix 49

From the Editor 50

Space for Notes 51

The 800- Plus 256K

Revision A: October 85
by David G. Byrd, KD7VA
Copyright (c) 1985, D.G.Byrd

Reprinted more or less intact from AD ASTRA, newsletter of the ATARI MICROCOMPUTER NETWORK, with permission of the author.

The following is provided free for the personal use of ATARI enthusiasts everywhere, but not for sale.

Parts List

Designation	Quantity	Type	Description
ZX1	1	74LS02	Quad NOR Gate
ZX2	1	74LS175	Quad Latch
ZX4-5	2	74LS158	Quad Multiplexer
ZX6	1	74LS112	Dual Flip Flop
ZX7-14	8	41256	DRAM (150ns)

Modification Instructions

Getting Started-

Remove the middle memory board from your 800, or use a spare, and take a look at it.

The eight chips along the top are the RAM chips. The other four chips are the addressing circuitry. The edge pin connectors at the bottom are labeled as in figure 2.

The first step is to eliminate the 12V and -5V sources on the board and to relocate the 5V source to where the 12V was. As shown in figure 3, cut the trace going from pin "X" of the board's edge connector to the capacitor C521. Also cut the trace going from edge pin "Y" to C523. Cut the traces cleanly and completely. Be careful not to slip and damage adjacent traces.

Now remove the capacitors C521 and C523. The trace coming from pin "W" carries 5V. Using a short piece of wire, make a solder bridge between this trace and the old 12V and -5V traces, at the point where C523 and C521 used to be, as shown in figure 3. Next, remove the eight capacitors, C503, C505, C507, C509, C511, C513, C515 and C517, which are usually in a row along the top of the board.

We now have 5V going to pins 1, 8 and 9 of the RAM sockets. Remove the eight 16K RAM chips and insert new 256K RAM chips in their place, properly orienting their notched ends. With an ohmeter, make sure there is NO connection between edge pin "Y" and pin 8 of the chips, nor should there be any connection between any two of the edge pins "W", "X" and "Y".

If all has gone well, the board should function like a regular 16K memory board, since the addressing circuitry has not been altered yet.

Replace the modified memory board in your 800's middle slot and power up your machine. The standard blue screen should appear quickly. If

not, recheck all modifications made so far.

The Next Step-

Turn your 800 off and pull your modified memory board from the middle slot again. Cut the two rightmost wide traces on the chip side of the board, as shown in figure 4. This removes the 5V supply from pins 1 and 9 of the RAM chips.

Pick up a new 74LS158 chip, which is the same as chips Z503 and Z504 on the memory board. With needle nose pliers, carefully bend up all pins except 1, 8, 15 and 16 as shown in figure 5. Remove chip Z503; place the new chip on top so that the four pins listed above touch the same four pins on the lower chip. Carefully solder each of the four pairs together, being careful not to get too much solder on the pin ends. (It still has to fit back in the socket!) Now insert the piggyback pair in socket Z503.

The second new 74LS158 chip is to be mounted in the same fashion atop Z504, EXCEPT that pin 1 is elevated and not soldered. Insert this piggyback pair in socket Z504.

Next, we will prepare the additional chips for mounting. These chips will be mounted "Dead Bug" style, i.e. "legs" up, so it is a good ideas to put a stick-on label on the underside of each chip indicating its type and the location of pin 1.

Install the three new chips in the vacant area below the memory chips. You should use "Super Glue" to attach them, so it is VERY important to get them in the correct place the first time.

Now wire the new chips per the diagram in figure 6. Connect 5V and ground pins first to each "Dead Bug" chip. Convenient connection points for 5V and ground are left and right ends of each capacitor located directly below each RAM chip. Connect each chip individually with short wires.

Connect a temporary jumper from edge pin "V" to "W".

Progress Check #2-

Return the modified board to the 800's middle slot for testing as a 16K board. Power up your machine. If the standard blue screen doesn't appear quickly, SHUT DOWN IMMEDIATELY and check your work.

If the blue screen does come up, shut down, pull the modified board, and remove the temporary jumper from pins "V" to "W". Cut the track connecting pins "S" and "T".

The 256K board is complete, so put it aside for now.

Standard Memory Board Mod-

Remove the remaining two 16K boards from your 800 and remove the cases, if they have them. On the reverse side of each board, find the tracks connecting pins "S" to "T". Cut this track on both boards. If you had cases, you can reinstall the boards in them. Set these two boards

aside.

Personality Board (ROM) Mod-

Remove the ROM board from its slot. Take it out of its case if necessary. Find the Z401 and connect a jumper between Z401 pin 7 and pin 20 of the card edge. (20 is the only unused pin.)- Do not allow solder to flow down the card edge pin. Reinstall the ROM board in its case, if it had one, and set it aside.

The Backplane Jumpers-

Now it will be necessary to get to the 800's backplane. This requires removal of the case. First, remove all cables from the computer. Turn the 800 over, with the i/o ports facing you, and remove the five recessed phillips screws holding the bottom cover in place. It may now be removed by lifting at the rear and then sliding it forward to clear the i/o port connectors. This will expose the bottom RF shield.

Remove the nine phillips screws holding the bottom RF shield and mother board in the RF cage. Remove the phillips screws on the sides of the RF cage. GENTLY lift the front edge of the mother board and power supply boards and disconnect the keyboard ribbon cable, the power supply cable (noting its orientation!) and the speaker cable. Separate the power supply board and mother board. The mother board and the lower RF shield may now be lifted up and out of the RF cage. Now remove the CPU card.

The bottom shield must be removed. It is attached with four plastic expansion pins. Use a screwdriver to push the center pin out until it can be removed from the bottom. Now remove the outer pins.

The bottom shield will lift off and expose the entire backplane area under the card cage.

Using figure 7 as a guide, install the three jumpers using a fine, solid-conductor, insulated wire.

Recheck your work with particular attention to correct pins, solder bridges and shorts. Then check it again. And again, just in case.

Reinstall the bottom shield and replace the plastic pins.

Reassembly-

Reinstall the CPU board, place the mother board in the RF cage, and re-insert the power supply connector, checking it for proper orientation. Reconnect the keyboard ribbon cable and the speaker cable. Replace the screws in the RF cage and then the nine screw holding the bottom shield. Reinstall the bottom cover and replace the five phillips screws. The 800 - PLUS 256K modification is now complete!

The Last Test-

Turn the 800 upright and reinstall all the cards you have set aside, making sure the 256K board is in the middle slot. Replace the top

cover and close the cartridge lid. Power up your machine. Again, if the blue screen fails to appear, IMMEDIATELY shut down and check your work.

If the blue screen does appear, congratulations! You now have an Atari 800 with 48K of normal RAM and 240K of extended memory.

Now What?-

So you've got all this memory. What do you do with it? Well, fortunately, there are a number of ways to access this extra space. Unfortunately, it isn't the same as a 130XE. The memory location the 130XE uses as the memory control register is used for i/o ports 3 & 4 in the 800. But the 240K can be used as a virtual disk (RAMDisk) to speed up disk operations when using non-boot software. So there are many applications and there is software to make them available.

Editor's Notes-

If you are interested, the author, Dave Byrd, does have a ROM modification he can do that will make the 240K an AXLON RAMDISK act-alike. And SynCalc and SynFile+ are AXLON compatible. That makes SynCalc a 240K spreadsheet instead of a 21K. And SynFile+ reports "288K RAM Available" when it boots up. That means a massive increase in record availablility. Dave Byrd's address is 1513 Commanche Drive, Las Vegas, NV 89109. This guy is a hacker of the highest order and worthy of any and all praise he gets. And he'll do the mod for you if you're a little squeamish about opening up your 800. His mod is plug-in, so all you need to do is send him your ROM board and a 16K RAM board and $70.00 and you'll be in RAM heaven in no time. (And if you've got DONIMON, it's no problem.)

He also has a number of DOS varieties, that initialize the 240K as various drives. And there are diagnostics tests, too. Write and ask for costs, etc.

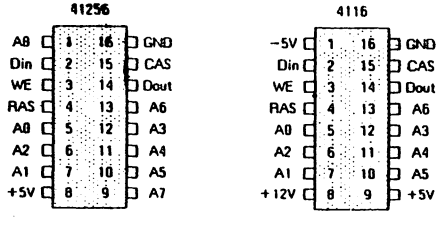


FIGURE 1 – Pinout Comparison of 256k and 16k bit RAM's

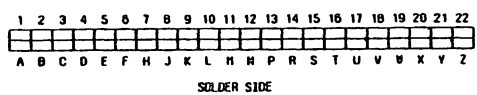


FIGURE 2 – Card Edge Connector Pin Assignments

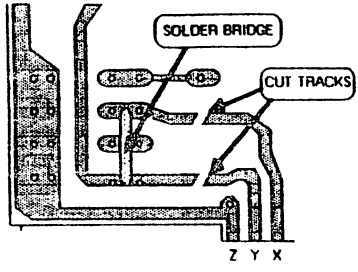


FIGURE 3 – Lower Left corner of Memory Board (Solder Side)

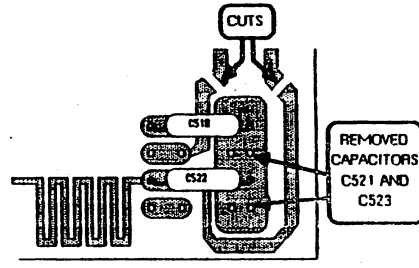


FIGURE 4 – Lower right corner of Memory Board (Component Side)

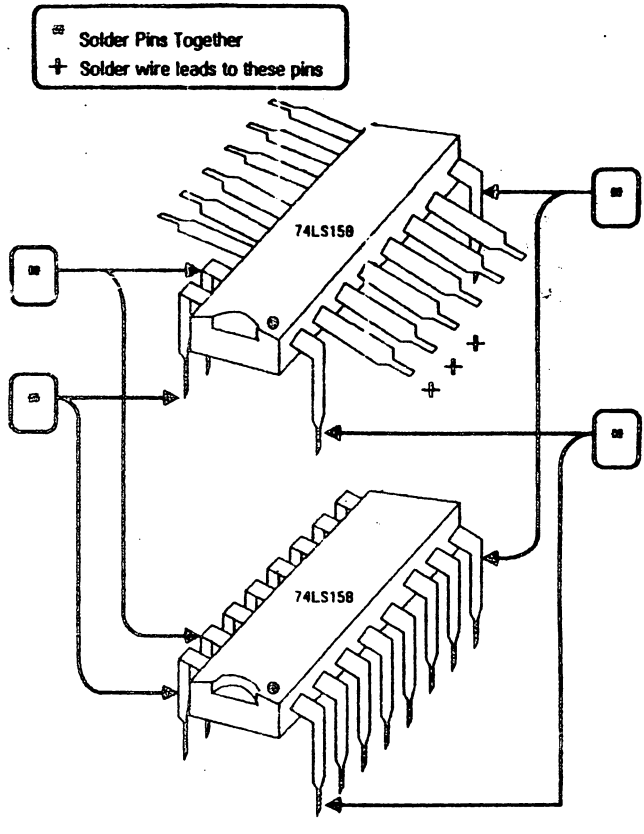


FIGURE 5 – Piggyback Arrangement

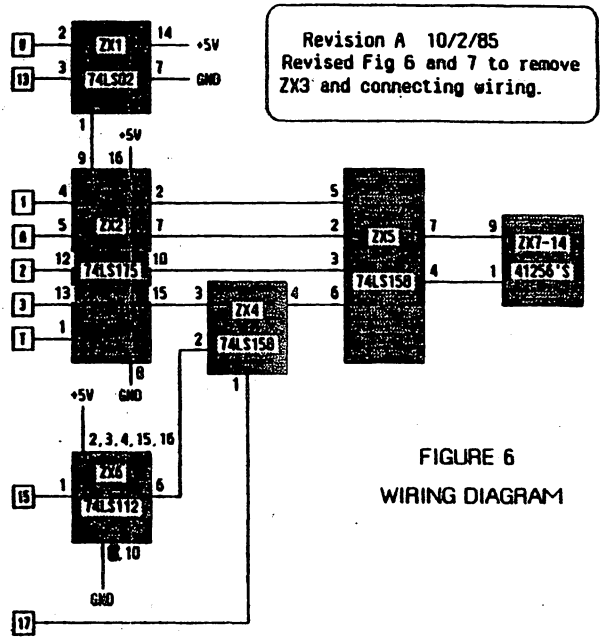


FIGURE 6
WIRING DIAGRAM

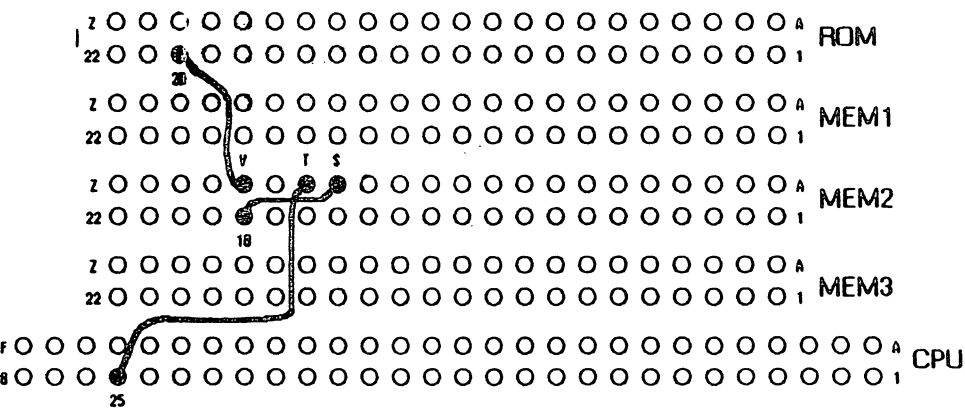


FIGURE 7
BACKPLANE JUMPERS

800+++++ 256-512-1-2-4 MEG!!

by Roger E. King, S.P.A.C.E.

First, credit where credit is due. D.G.Byrd must be recognized for his truly logical approach to more memory by bankswitching the 800. David Lloyd brought us AXLON compatibility and Kurt Grittener and his MAAUG members showed us the XLs could support more than 250K. Terry Shingara and Steve Jones were my cohorts in simplifying installation and redesigning areas to make them more concise and compatible.

In order to make the original 256K mod a one-board affair, the following was accomplished. (See figure 6.) "Reset" from the processor board was replaced with a resistor/cap circuit. Pin "18" to pin "S" is now a ramboard jumper. "D7XX" is now a board-to-board jumper. That's all, folks..... for the 256K portion of our update.

For Axlon compatibility, this circuit needed a little help, as it hung on programs requiring warmstarts. The new design (figure REK 7) supports "D7XX" and AXLON very well. (This circuit will falter if "D8XX" is accessed, which is very unlikely.)

More than 256K on your 800???? YOU BETTER BET'CHA! The 2 to 4 MEG upgrade has been in use since January of '86. This mod (figure REK 8) uses D.G. Byrd's "D7XX", the data going to "D7XX" and a signal called "CAS." In the REK upgrade, using the data MBD0 through MBD6 will enable the addressing of up to 2 MEG. Using MBD7 allows the addressing of an additional 2 MEG. Theoretically, the "D7" decode is being used to gate the addresses. If the "XX" portion was used, there would be an additional 00-FF (0-256) gates. This could mean 255 x 4 Meg. (That's 1020 Megabytes!) This is a lot more ambitious than most people are willing to attempt, but it gives a glimpse of the hidden powers <u>STILL</u> available for future enhancements.

The information for these upgrades and mods are presented as is... FREE. The author would like to see them put to both efficient and imaginative use, so have at it!

There is a modification to SmartDos, available from JonesWare, that allows
1-2 RAMdisks, SS or SD, and Archiver, allowing one-pass copying. These mods recognize Axlon/D7xx/D301 addressing schemes.

Roger E. King, S.P.A.C.E., P.O. Box 1446, Harrisburg, PA 17106-1446

Steve Jones, JonesWare, P.O. Box 7037, Mechanicsburg, PA 17055-7037

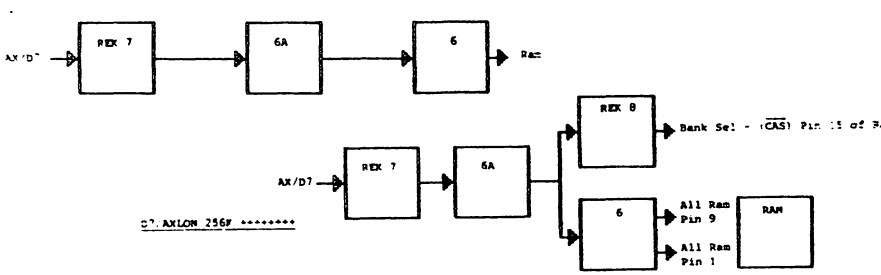


Figure REK 7
!@n/D7xx Wiring Diagram
(Piggyback this chip onto Z402 of personality bd w/pins 7,14)

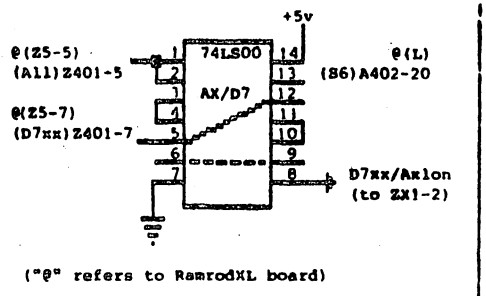
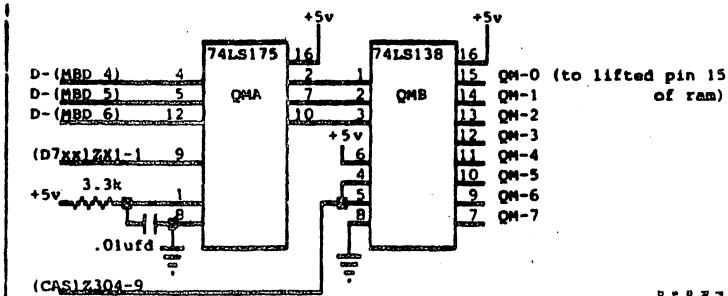


Figure REK 8
256k-4MEG Diagram



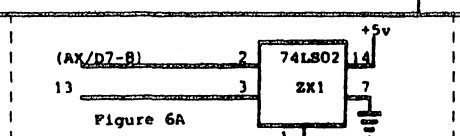
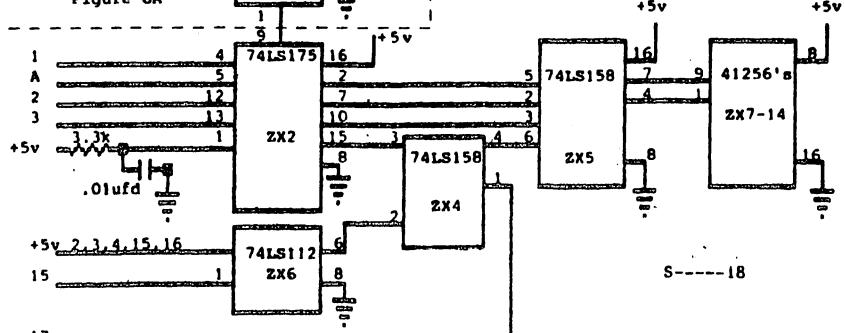


Figure 6
Modified 256k Diagram



NOTE: Unlike David Byrd's 256K, all wiring is on the ram and personality boards with one jumper between them to provide "D7xx and/or Ax:oc". Figure 6 may be used alone with jumper to Z401-7 for 256K. Figure REK 8 may be used to enable from one to 8-256K banks of additional memory. And lastly, Figure REK 7 can be implemented with either setup to provide Axlon/D7xx usage. To go beyond the 8 additional banks, see MBD with circuit outside in Figure REK8. GOOD LUCK.

REK

Atari 800XE Modification

by Gary Monti

(The following article is a somewhat abridged version of the original that ran in HACK, the newsletter of the AAUG of RI. Before you start, read the whole thing twice.)

The first step is determining which video controller your 800XL has. This is accomplished by taking apart the machine so you can see the motherboard. BE CAREFUL! After removing the six screws on the bottom, carefully separate the top and the bottom. The keyboard ribbon cable is fragile and must be pulled straight up out of its socket. Next, remove the three screws holding the motherboard to the bottom case and take it out. Then remove the small nuts and bolts securing the top shield and remove it. You should now be able to find the video controller chip. It is a 40 pin chip, U7, and at the bottom center of the center card. If it has C021697 stamped on it, you can use the simpler circuit, eliminating the circuitry in the dotted lines of the schematic. If it is stamped C012296, you'll have to use the extra circuitry.

Parts List

Quantity	#	Description
8	41256	256K Dynamic RAM (200ns)
1	74LS153	Dual 4-to-1 multiplexer
1	74LS139	Dual 2-to-1 Decoder
1	-	33 Ohm, 1/4 watt resistor
9	inches	16 wire ribbon cable
2	-	16 pin DIP header connectors
1	sm. spool	30 awg. solid wire
7	-	P.C. sockets
1	-	I.C. breadboard Radio Shack 276-150

Additional Parts for C012296 owners

1	74LS158	Quad inverting 2-to-1 multiplexer
1	74LS393	Dual 4-bit counter

You'll also need the following tools: Soldering iron, desoldering tool, small diagonal cutters, long nose pliers, wire strippers and an ohm meter.

Wiring Instructions

The first thing to do is construct the cable. Take the ribbon cable and attach a DIP header connector to each end. Pin 1 will be in the upper left corner; mark it. Find out what pin it corresponds to on the other DIP connector and mark it Pin 1 also. Assemble your sockets on the breadboard, four for the simpler circuit, six for the more complex. Plug one side of the ribbon cable DIP header into one of the sockets on the end of the breadboard. The cable will connect to U27 on the Atari motherboard, so the socket can be designated as such. Mark the remaining sockets per the schematic. Begin your wiring from U27 to IC1 (IC socket 1). Check your connections with the ohm meter as you go. Make sure the pins match up, too! Wire the rest of the breadboard according to the schematic. When you place the wires for the jumpers that go off the breadboard, leave about 15 inches slack. Take an extra socket and cut out a group of five pins. To this group of pins, attach

a set of jumper wires that will go to U23 on the motherboard per the schematic. Make sure you attach them in order. Now, check all your connections with the ohm meter, especially for shorts between power, and the jumpers to U23.

Installation

Back on the motherboard, you'll find a row of eight 16 pin RAM chips. Directly to their right is U27. It is either 74LS158 or 74LS258. When you find it, remove it. Use the desolderer if necessary. Now, install it in IC1 on the breadboard, making sure the pins line up where they should be. Take the loose end of the ribbon cable and plug it in the now empty U27 socket on the motherboard, again making sure that Pin 1 is where Pin 1 should be. Check the circuit.

At the top of the RAM chips, you'll find a resistor marked R32. Remove it. One of these holes runs to pon 1 of the RAMs. It's usually the one to the inside. Solder your first jumper here, the one from IC4 pin 1.

Locate U23, a large 40 pin chip at the right bottom of the motherboard. Carefully pry it from its socket. Use your long nose pliers to bend out pins 12 through 16, so they stick straight out to the side. Reinsert the chip. Take the five jumper wires from the breadboard and insert them in the designated socket holes. Check your circuits. Place some electrical tape around the connection. If you are using the complex circuit, you'll need a fifth jumper. Locate U7 pin 8. Locate the etch on the motherboard that runs from pin 8. At one point on that etch, you'll find a soldered over hole. Solder the jumper from IC4 pin 1 to that hole. Now, install the chips in their proper sockets, making sure they're lined up correctly.

The next step is to carefully remove the 64K RAM chips and replace them with the 256K RAM chips. These chips are VERY sensitive to static electricity, so BE CAREFUL! The last thing to do is insert a piece of plastic or cardboard under the breadboard to prevent it from shorting out. Finally, replace the shielding around the motherboard, bending it a little if necessary for a fit, and reassemble the computer in its case.

Debugging

If all has gone well, the computer will act just like it did before you did the upgrade. Power up and check it out. If you don't get a regular blue screen, SHUT DOWN IMMEDIATELY and recheck all your work. If you get the blue screen, boot up a DOS 2.5 disk and wait. If it says "Atari 130XE Ram Disk Loading - Please Wait," rejoice and relax. You done good.

Credits

Claus Buchholz, of Lansing, MI, is the designer of this upgrade, a true 130XE compatible modification, and has done so, at the expense of much of his time, with no expectation of monetary gain. All he wants is to have the information passed along between user's groups, so that software for its use may be developed. Hats off to him! Luis Collado, Boston Computer Society member, breadboarded the project and answered

an avalanche of questions. Without his help and ear, the debugging of this project would still be going on. Hats off to him, too!

Souping Up the 1200XL

by Paul Smith

Those of you who like your 1200XL may have felt a little forlorn recently with all the nifty mods available for the other XL and XE machines. Well, help is on the way. I've had a chance to tear apart a 1200XL and "make it all it should have been." I'll describe the various mods available for those of you with enough know-how (or enough guts) to try.

Printer Interfaces: Some of the "Apeface" style of printer interfaces won't work on the 1200XL. This is because they get their power from the computer. The power available from the 1200XL is not enough to drive these external devices. To fix this, short out R63. This is the 100 ohm resistor (brown-black-brown) in series with the power supply and is located near the serial I/O connector (J1). Just solder a wire from one end of the resistor to the other. It is easiest to do this from the solder side of the board.

Serial I/O Connector: Connect together pins 4 and 6 of the serial I/O connector. These should be both connected together and then to ground. The 1200XL schematic shows these both connected, but on my machine, only one was. As before, this is easiest to do on the solder side of the board.

Chroma Output: As with the entire XL series, the chroma output has been left off of the monitor jack. This can be restored quite easily with the addition of a single 75 ohm resistor (violet-green-black). Find the transistor Q7. Its emitter is soldered to the square pad (the others are circular). On the solder side of the board, connect one end of the resistor to this square solder pad. Connect the other end of the resistor to the unused pin of the monitor jack (J2) through a wire of appropriate length.

XE-Compatibility: An XE-compatible 256K upgrade is possible. The illustration illustrates the schematic for this mod. To perform this mod, remove R30 (near DRAM U5). Also lift up pins 12 and 13 of the PIA (U23) from the socket (with this mod, you lose the use of the two LEDs on the front panel). Remove the 8 DRAMs located along the front row of the PC board. Build up the circuit shown in the illustration. The parts are:

UA	74LS393
UB	74LS153
UC	74LS151
UD	74LS10

Since the shield is so low, "piggyback" technique can't be used. Nor is there much room for a small PC board. I hand wired the parts by placing the ICs upside down ("dead bug style") on the top side of the PC board and soldering from leg to leg. When connecting the circuit, Pin 1 of the DRAM is hole left by R30 nearest DRAM U5. Lift pin 9 from U10. This used to be connected to one end of R97. Now this end of R97 should be connected to pin 6 of the new 75LS151 (UC) To test this mod, run any of the programs written to utilize the 130XE's extra memory. Or better yet, try running on of the RAMdisk programs available for the 256K 800XLs.

Translator: It is possible to re-program the OS sockets to accept 2764 EPROMs. This will allow you to have built-in translators installed in your machine. To perform this mod, locate jumpers W6 through W13. They must be set up according to the following tabel (note W10 is not used for this - it is used to program the GTIA for US or European TV standards):

JUMPER	ROMS	EPROM
W6	IN	OUT
W7	IN	OUT
W8	IN	OUT
W9	IN	OUT
W11	OUT	IN
W12	OUT	IN
W13	OUT	IN

With these mods installed, you have brought your machine up to the state-of-the-art (although it is still a little large). It should be able to take on all comers.

Reprinted courtesy of the North Texas Atari Computer Team.

130XE/320K Upgrade

by Scott Peterson.
(modified by Tom Harker of ICD 1/25/86)

After reading and building both the 800/288K upgrade (D.G.Byrd), and the 800XL/256K upgrade (C.Buchholz), I decided that there had to be a way to upgrade the 130XE. There is, and thanks to the "Freddie" chip (C061991) this modification was much easier to design than either of the other upgrades.

Since all ICs are soldered directly to the board on a 130XE, installation requires excellent soldering and de-soldering skills.

Tools

To install this upgrade you will need: a #2 phillips head screwdriver, small needle nosed pliers, a low wattage soldering iron, a de-soldering tool, and some fine (30 gauge) hook-up wire.

Parts List

1 - 74LS158 (2 to 1 Multiplexer)
8 - 41256 256K dynamic RAM (150ns)
1 - 33 ohm 1/4 watt resistor.
8 - 16 pin low profile sockets.

Doin' it

First, remove both the case and the metal shield to get down to the mother-board. Then de-solder and remove the eight ram-chips U26 thru U33 (MT4264). They are the inside row (closest to the TV RF module). Next, replace these with sixteen pin low profile sockets. Take your new 256K DRAMs and bend pin 1 straight up on each chip. Plug these chips into the sockets you just installed. Now take a piece of wire approx 12 in. long and solder a jumper from pin 1 on each of the 256K ram-chips so they are all connected together.

Next, cut and bend up pin 15 on U23 (C014795 the PIA) or instead, remove R206 (closest to pin 20) and cut the foil trace running from pin 15 towards the front of the board. Take your new 74LS158 and break off pins 5,6,7,9,10,11,12,13,14. Bend up the other pins except for pins 8 and 16. Install this "piggy-back" on top of U20 (HD14050 - located just to the right of C50) and solder pins 8 and 16 of this 74LS158 to pins 8 and 16 on U20. Now connect pin 15 (74LS158) to pin 8 (ground) with a short jumper wire.

Take a piece of wire about 4 in. long solder one end to pin 30 on the chip marked "C014805" (this is 82 on the GTIA) on the mother board, and the other end to pin 1 of the new 74LS158. Next, solder a wire to pin 15 (the one you bent out) of U23 (PIA) and connect the other end to pin 2 on the new 74LS158. Solder a wire to pin 16 on U23 and connect the other end to pin 3 on the new 74LS158.

Take a 33 ohm 1/4 watt resistor and trim the leads to about 1/4 in. Solder one end of it to pin 4 of the new 74LS158. Connect a wire from pin 1 of the new DRAMs (any one of them) to the other end of this resistor. Re-assemble the RF shield making sure you are not shorting it to any of the modified hardware. Assemble the case and you are finished!

Usin' it

The best deal for this mod is to use SpartaDOS (available with SpartaDOS Construction Set or the US Doubler). The new RD.COM file supports it as a full 256K RAMdisk (any drive number) or a 192K RAMdisk with 64K reserved for BASIC XE. This 320XE modification is also totally compatible with ICD's RAMBO XL mod. for the 800XL and 1200XL computers. Questions on SpartaDOS with this mod. and RAMBO XL will be answered on the ICD BBS 815/968-2229, 24 hours, 300/1200/2400 baud.

MYDOS 4.0 also supports a very large single density RAMdisk. With BASIC XE you can use a 1500 sector RAMdisk and without it you can have about 2000 sectors.

If you are a hot-shot programmer (I'm not) I think a print spooler that uses part of this RAM would also be very nice. This mod is easy to do and perfect for running a BBS. One note, on COMPUSERVE, there is a mod by Rich Andrews which should not be confused with this one. His uses 33 new chips and mine uses 9 new chips.

320XE Memory Control Register 54017 ($D301)

```
     Bit 7 6 5 4 3 2 1 0
         D a b C c d B R
```

D=0 enable diagnostic ROM.
B=0 enable BASIC ROM.
R=1 enable OS ROM.
C=0 enable extended RAM.
abcd= memory control bits.

Bank #	Control#		
Bank 0	)131		
Bank 1	)135		
Bank 2	)139		
Bank 3	)143		
Bank 4	)163	Basic=	off
Bank 5	)167	OS =	on
Bank 6	)171	ENH =	on
Bank 7	)175		
Bank 8	)195		
Bank 9	)199		
Bank 10	)203		
Bank 11	)207		
Bank 12	)227	<--\	
Bank 13	)231	\ 130XE	
Bank 14	)235	/ Banks	
Bank 15	)239	<--/	

If you are using MYDOS 3.016 and wish to use BASIC XE and a RAMdisk at the same time, boot DOS and poke 5275,163 and 5324,16. Go to DOS and write the new DOS. This will keep the two from "bumping" into each other. A similar poke can be done to DOS 2.5, it is poke 4838,163. The handler I have will set up 192K of the extra ram as 2 SD RAMdisks or 1 DD RAMdisk.

This upgrade has been built and tested on a BBS. It has run for days on end without a memory loss or error. If you need help or more information feel free to call the Peanut Gallery (408)-384-3906. 24HR, 300/1200 Baud. Leave mail to the Sysop (that's me). Good luck and let me know if you write a better handler for DOS 2.5. by Scott Peterson

Where Did All This Come From?

The members of MAGIC didn't write these articles. We just collected them. Mostly from newsletters put out by Atari Users Groups. If you aren't involved in a users group now, you're probably missing something. The users group is rapidly becoming the only information pipeline for 8-bit Atari owners. And unfortunately, many 8-bit Atari owners are dropping out of users groups all over the country, because of the influx of ST owners, or some other silly reason. So for many 8-bit Atari owners, there is a lonely, lingering death for their machine. And a very premature death it is. The full power of 8-bit technology is yet to be realized. And the people who are pushing toward that goal are the diehard users group members who don't let the new kids on the block push them out of their own clubs. The dollar Atari computers are wonderful machines to start with, and with a little effort, they can be phenomonal little things, able to leap large spreadsheets in a single bound. Maybe they're not as fast as the ST, but most of the time, a VW bug gets you there just as well as a Corvette.

So what's this all about? Users Groups. Join one and join a bunch of people like yourself, who've selected the best little computer for the money around. And who just may have some tricks up their sleeves to make that bargain even better.

What if you don't know where the nearest users group is? Send a note or call Atari. They would like you to send your User Group List requests to Atari User Group List, P.O. Box 61657, Sunnyvale, CA 94088. The generic phone number for Atari is (408)745-2000. It'll probably take a couple weeks to get a reply, so hang in there. And keep in mind that the list may not be too current. Groups come and go, so you might want to try more than one.

Another place to look for a users group is in the various magazines that support the Atari computers. Computer Shopper has a listing of groups in every issue. ANTIC and ANALOG run small list pretty much every month, too.

But wherever you find a group, join it and get involved. Do a demonstration of a program. Run for a club officer position. Write an article for the newsletter. Anything. The more you put into a club, the more you get out of it. Don't be afraid to sound like a rookie. Everybody is at first. And the only way to get beyond that is to ask questions and talk to those who have gone before you. It's relatively painless, and infinitely rewarding.

Have fun and get involved. You'll be glad you did!

130XE/576K Upgrade

by Scott Peterson.

Here we go again, this time I recommend you have some electronics experience if you wish to perform the upgrade. Some of the work is duplicated from the 320K upgrade so 320XE owners will not have as much work to do. One other point, when in the 576K mode, you MUST use some sort of BASIC cartridge, as you lose the internal basic. This is only in the 576K mode. In the 130XE mode, the BASIC will operate normally.

Tools needed:

a #2 phillips head screwdriver,
a pair of small needle nosed pliers,
a low wattage fine tip soldering iron,
a vacuum de-soldering tool, such as Radio Shack #64-2098
some fine (30 gauge) hook-up wire, Radio Shack #278-501,
some heat-shrink tubing, 1/8 inch diameter,
a small flat tip screwdriver.

Parts needed:

Z1	74LS158
Z2-Z17	41256 (150ns)
Z18	74LS138
Z19	7432
R1-R2	33 ohm 1/4 watt resistor
S1	micro-mini DPDT switch, Radio Shack #275-626

First, remove both the case and the metal shield to get down to the mother-board. 320XE owners may skip to step two.

Step One:
Now de-solder and remove the eight ram-chips U26 thru U33 (MT4264). They are row closest to the TV RF module. (Do NOT use solder wick, as the circuit board of the 130XE has very weak runs and will pull loose if not completely de-soldered.) Next, replace these with sixteen pin low profile sockets. Take a piece of wire approximately 12 inches long and run a jumper from pin 1 of each socket to the next. When you are done, the wire should be attached to pin 1 of each of the new sockets and you should have about 6 inches of wire left over. Do this on the rear of the mother board and then snake the wire through the large hole near the RAM chips.

Next desolder and remove U23 (CO14795) and replace it with a 40 pin socket. Bend up pins 15 and 16 on U23 and re-insert it in the socket you just installed. Take Z1 (74LS158) and break off pins 5, 6, 7, 9, 10, 11, 12, 13 and 14. Bend up the other pins on it except 8 and 16. Put this "piggyback" on top of U20 (HD14050, or 4050 - located just to the right of C50) and solder pins 8 and 16 of Z1 to pins 8 and 16 on U20. Now take a short jumper from pin 15 on Z1 to pin 8 of Z1. Take a piece of wire about 4 inches long, solder one end to pin 30 on the chip marked "CO14805" on the mother board, and the other end to pin 1 on Z1. Next, solder a wire to pin 15 (one of the two you bent out) of U23 and

connect the other end to pin 2 of Z1. Solder a wire to pin 16 on U23 (the other one you bent out) and connect the other end to pin 3 of Z1

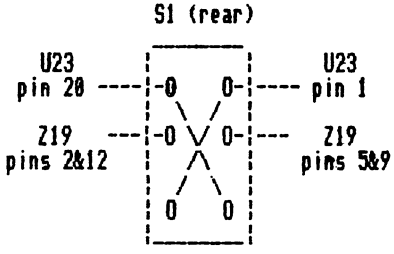
Take R1(33 ohm) and trim the leads to about 1/4 in. Take the wire you connected to pin 1 of the RAM sockets and solder it to one end of R1. Solder the other end of R1 to pin 4 of Z1.

Step Two:
Slide the mother board back into the bottom half of the plastic case (do not use the RF shield, as you must be able to get to the mother board), and attach the keyboard. It will rest above the mother board without touching it. Test all the 41256 RAM chips by putting one set of 8 in the sockets and using the handlers (or DOS's), and then the next. After testing all the RAM chips, remove them all from the sockets, and take 8 of them and cut about half of pin 15 off of each one. Only the "fat" part of pin 15 should be left. After doing this, you have to "piggyback" the 8 256K RAM chips with the short pin 15's on top of the other 8 256K RAM chips. Now solder all the pins together on the stacked RAM chips except for pin 15. It should NOT be touching the other pin 15. Make sure you have pin 1 to pin 1, pin 2 to pin 2, etc. When you get done, you will have 8 sets of "piggybacked" 256K RAM chips. Now take a piece of wire about 16 inches long and run a jumper from pin 15 to the next pin 15 on all the top 256K RAM chips, leaving about one inch between each "piggybacked" chip. Put these stacked RAM chips into the 8 sockets you installed earlier. Take Z18 (74LS138), bend up all the pins except 8 and 16, cut the pins you bent up in half so only the "fat" part is left, and solder pins 8 and 16 to pins 8 and 16 of the other 74LS138 right below the U23 (CO14795). Take Z19 (7432), bend up all the pins except 7 and 14, cut all the pins you bent up in half so only the "fat" part remains and solder pins 7 and 14 to pins 7 and 14 of the 74LS508 right below the U23.

Take the wire you jumpered to pin 15 of Z10 through Z17 (the upper RAM chips) and go out 2 inches. Cut the wire and install R2 (33 ohm) between this cut. Place a piece of heat shrink tubing over R2 and make sure no wire is exposed and heat it with a lighter. Take the other end of this wire and connect it to pin 14 of Z18. Find the 2 33 ohm resistors just to the right of U28 (one of the RAM chips you socketized). The upper one of the 2 is R111. Desolder the right leg of it and bend it up. Take a piece of wire and solder it to the land where you just removed the leg of R111. Connect the other end to pin 12 of Z18. Take a short wire and run a jumper from pins 1 and 16 of Z18. Take another short wire and connect a jumper from pins 3, 5 and 8 of Z18. Now connect a wire from pin 2 of Z18 to pin 3 of Z19.

Find the wire you installed from pin 15 of U23 to pin 2 of Z1 (74LS158) and desolder it from U23. Take it and reconnect it to pin 11 of Z19. Okay, now pry U23 (CO14795) back out of the socket, bend up pin 11 and plug it back in. Run a jumper from pins 1 and 4 of Z19, and another jumper from pins 10 and 13 of Z19. Connect a wire from pin 11 of U23 to pin 1 of Z19, and from pin 15 of U23 to pin 13 of Z19. Connect a wire from pin 8 of Z19 to the right side of the 3.3K ohm resistor marked R206 (located at the bottom right of U23). Connect a wire from pin 6 of Z19 and run it to pin 18 of U3 (C)61618). Now comes the tricky part. Drill a small hole (1/4 inch or so depending on the size of your S1 switch) at the right rear on the back of your 130XE case. Take S1 and install it in the hole. Now connect it as shown (make sure

the switch doesn't have a center off position).



Note: Where the wires cross in the middle, they are NOT connected. Make the connection from the rear of the mother board. Well, that's it (thank god). Now re-assemble the computer, being careful not to break any wiring going out to the switch. You should now have in one switch position a 100% compatible 130XE, and in the other you have a 576K 130XE that does not have Antic memory enhance mode and also connot use internal BASIC. In the 130XE mode, you gain 64K, as bit 6 of the PIA can still be used. The following is a list of the bit table and numbers to be used in location 54017 (PORTB).

Once again, if you need help, call the Peanut Gallery BBS (408) 384-3988. If you want a mailer of all the upgrades I have, as well as a disk with handlers, source codes, etc., send a money order (please, no checks) for $10.00 to:

Scott Peterson
P.O. Box 33
Ft. Ord, CA 93941-0033

XEMEG- The 1 Meg 130XE

by Pete Pacione and Thad Mack

(Reprinted from the CL.A.U.G. newsletter.)

I've been told that there is a right way and a wrong way of doing things. In most of our projects, we've always wanted to retain compatibility. So when thinking about enhancing the RAM in the Atari 130XE, our goal was not to change what already is a good machine, but to add to its capabilities. After all, there are quite a few programs written for the 130XE using its extended RAM.

Now, I'm aware of the modifications that already exist for the 130XE to extend RAM. First of all, you lose Video Bank Enable (VBE), internal BASIC is removed and the Operating System must be permanently turned on. This means that the Translator Disk cannot be used. Of course, you could install an external switch, to turn the OS on or off; however, I never did like drilling holes in my equipment. And why do something manually, when it can be done automatically? I guess this goes back to the right way and wrong way of doing things.

The standard area of extended RAM switching on the 130XE is memory location $D301 (hex). There are not enough free bits at this location to allow for more than 128K of extra memory. The logical solution... use another memory location! Ahh, but then it's not compatible! Not to worry... the most optimum memory location is $D700 (hex). At $D700, if we manipulate bits 1 and 2, we can manipulate one of four 16K banks. Bits 3 and 4 select one of four 64K banks, and bits 5 and 6 will select four 256K banks. This is done by writing a 00, 01, 10 or 11 in bit groups 1 and 2, 3 and 4, and 5 and 6. Bit 7 controls the extra 64K. The last bit, bit-8, is the eXtra CPU Bank Enable (XCBE). This bit, when 0, allows access to the extra memory. A 1 reverts the 130XE back to the original memory configuration. What we've done is, at power-up, the circuitry gives you a standard 130XE capable of running anything written for it. But when using a standard MYDOS version 4.2, you will have a 1 Meg RAMdisk. This gives 8170 single density sectors. The full upgrade, 1.064 Meg, will give you 8682 sectors! It's all automatic, the switching and everything! Once you have the modification installed and have purchased MYDOS version 4.2 (or greater), run the program at the end of this article and you can have a RAMdisk of 1.064 Meg. This article will allow you to install, in your 130XE, extra RAM up to 1.064 Meg, which is the upper limit at this time. But depending on your finances, you could install any amount from 256K to the limit, in 256K increments.

We have found the approximate cost of installing the 1.064 Meg at about $100.

WARNING... BEFORE YOU START THIS UPGRADE, BEWARE!!
YOU MUST HAVE GOOD SOLDERING AND DE-SOLDERING TECHNIQUES!!
THIS SHOULD NOT BE ATTEMPTED BY ANYONE THAT IS NOT
EXPERIENCED IN THESE SKILLS!! DAMAGE MAY RESULT WITHOUT
THE ABOVE SKILLS!! DO NOT USE ANY FORM OF SOLDER-WICK
OR BRAID FOR DE-SOLDERING!! ONLY USE A DE-SOLDERING PUMP
OR A SOLDERING IRON WITH A VACUUM BALL. ANYTHING ELSE
WILL DESTROY THE BOARD OR THE CHIPS OR BOTH!!

ALSO, FOLLOW NORMAL STATIC PREVENTION PRECAUTIONS!!

IT IS HIGHLY RECOMMENDED THAT YOU READ THROUGH THE ENTIRE ARTICLE BEFORE ATTEMPTING THIS MODIFICATION!!

Available soon will be a printed circuit board version of this modification. The board will hold all of the supplemental chips and wiring to keep construction time down.

Parts List

Quan.	Part #	Description
32	41256-150	256K x 1 150ns. DRAM
1	CD4030	I.C.
1	CD4042	I.C.
1	CD4053	I.C.
1	74LS00	I.C.
1	74LS02	I.C.
1	74LS138	I.C.
1	74LS158	I.C.
1	1 mfd 10v.	Capacitor (Electrolytic)
1	3 mfd 10v.	Capacitor (Electrolytic)
2	470 ohm	Resistor (1/4 watt)
5	33 ohm	Resistor (1/4 ohm)

Even though there are 32 DRAMs listed, you do not need to install all 32. If you only want 256K of extra memory then you will need only 8 DRAMs, 512K is 16 DRAMs. The amount is always in multiples of 8.

Building Instructions

Disassemble the 130XE down to the board level. If installing the full 1.064 Meg, the first thing to do is to carefully de-solder and remove both sets (16) of RAM chips (U9 thru U16 and U26 thru U33). Put these chips on the side in some kind of static safe package, as they will be used later. If installing less than the full Meg, remove only the Auxilliary Memory Bank (U26 thru U33). SEE DIAGRAM.

The first set of eight 256K DRAMs will be left as is. Take the second set of eight 256K DRAMs and bend pin 15 straight out on each of them. Then piggyback these over the first set. Leave an air gap between DRAMs (about 1/16th of an inch). Take the third set of eight 256K DRAMs and bend pin 15 straight out on each of them. Then piggyback these over the first and second sets. Trim the bent-out pin 15s, so that 1/8th of an inch remains. Set these aside, again in a static safe place.

Take the last set of eight 256K DRAMs and bend pin 15 straight out on each of them as before. Piggyback this set over one set of the 64K DRAMs you removed from the board. Take the other set of eight 64K DRAMs previously removed from the board, and bend pin 15 straight out on each of them. Piggyback these over the fourth set of 256K DRAMs (which are piggybacked over the first set of 64K DRAMs). Set these aside with the other static safe DRAM assembly.

Take the first DRAM assembly and install and solder them in the Aux Memory Bank (U26 thru U33). You now have eight vertical columns of

three chips. The chips that are soldered to the board are now called row one, the second set of chips up are row two, and the top set is row three. Using 30 AWG wire-wrap wire, connect pin 15 on all chips in row two together, and then repeat with pin 15 on all chips of row three. Solder a 4 inch length of wire from pin 15 of row two, by U33, and a 4 inch length of wire from pin 15 of row three, by U33. These wires will be attached later. The wire coming from row two is RAM Bank 1, and the wire coming from row three is RAM Bank 2. On the foil side of the board, wire pin 1 of all the row one chips in this assembly group together.

Next, install and solder the other assembly group in the Main Bank (U9 thru U16). These chips are thought of in the same way as the other DRAM assembly. Row one is the bottom set, row two is the middle set and row three is the top set. Wire pin 15 of each chip in row two together, and repeat with the chips in row three. Solder a 4 inch length of wire from pin 15 of row two, by U16, and a 4 inch length of wire to pin 15 of row three, by U16, also. These wires will be attached later. The wire coming from row two is RAM Bank 3 and the wire coming from row three is RAM Bank 4. On the foil side of the board, wire pin 1 of each row one chip of this DRAM assembly together.

Solder a wire from pin 1 of the Main Bank of DRAMs to pin 1 of the Aux Bank of DRAMs. Try to keep all wires as short as possible. Refer to drawings to install other ICs. On all pins that are bent out, trim the pins down to 1/8th inch. In ALL instances, ALL pin 1's face to the left!

On the foil side of the board, cut the foil trace between R23 and U34-pin 3, between R24 and U34-pin 4, between R25 and U34-pin 5. On the component side of the board, cut the foil trace from the right side of R111. Take the 74LS138 and bend pins 1, 2, 4, 6, 7, 9, 10, 11, 12, 13, 14 and 15 straight out. De-solder any solder that may be in the holes of unused location U35. Trim pin 8 of 74LS138 to 1/8th inch. Solder the 74LS38 into location U35. Pin 8 will be soldered to the foil. Next, take the 74LS158 and bend out all of the pins except pins 8 and 16. Trim pin 8 to 1/8th inch. Remove any solder from pin 16 of unused location U21. Install and solder the 74LS158 to location U21. Pin 8 will be soldered to the foil.

Take the CD4053 and bend out all pins except pin 16. Piggyback it to U03 and solder pin 16 to pin 16 of U03. Solder a short jumper wire from pins 6, 7 and 8 of the CD4053 to pin 10 of U03. Take one of the CD4042s and bend out all pins except pins 8 and 16. Piggyback it to U20 and solder pins 8 and 16 of this CD4042 to pins 8 and 16 of U20. Take the second CD4042 and bend out all the pins except 8 and 16. Piggyback onto U02 and solder pins 8 and 16 of this CD4042 to pins 8 and 16 of U02.

Take the 74LS02, bend out all the pins except 7 and 14. Take pin 7 and carefully bend sideways to the back of the IC. This is to allow pin 7 to be soldered to pin 8 of U24. Piggyback and solder pins 7 and 14 to pins 8 and 16 of U24 (7 to 8 and 14 to 16). Take the CD4030, bend out all the pins except 7 and 14. Take pin 7 and bend sideways toward the back of the IC. Piggyback and solder pins 7 and 14 to pins 8 and 16 of U25 (7 to 8 and 14 to 16).

Now wire the ICs per the diagrams, using the wire-wrap wire (30 AWG).

NOTE: If you are using the 64K DRAMs that were removed to gain the additional 64K, this bank must be the last bank used on the 74LS138. Example: If you have 256K plus 64K of Aux Bank, pin 15 from the 256K DRAMs would go to Bank 0 and pin 15 of the second 64K DRAMs would go to Bank 1. If installing 512K of Aux RAM, 256K DRAMs would go to Banks 0 and 1, and the 64K DRAMs would go to Bank 2 (and so on). This is very important! The modification will not work correctly with the installation program if the original 64K DRAMs are not the last, or highest, Bank selected.

The MYDOS Mod Program

The accompanying BASIC program will set up your RAMdisk to your specifications and modify MYDOS to reflect these changes and initialize the RAMdisk.

Authors' Note:

If you have any questions regarding any of the information presented in this project, feel free to leave a message on CL.A.U.G.'s BBS (312) 889-1240 or write to:

Pete Pacione
c/o CL.A.U.G.
2952 N. Meade Ave.
Chicago, IL 60634.

MyDos can be obtained from:

Wordmark Systems
2705 Pinewood Dr.
Garland, TX 75042

The price is around $ 39.95.

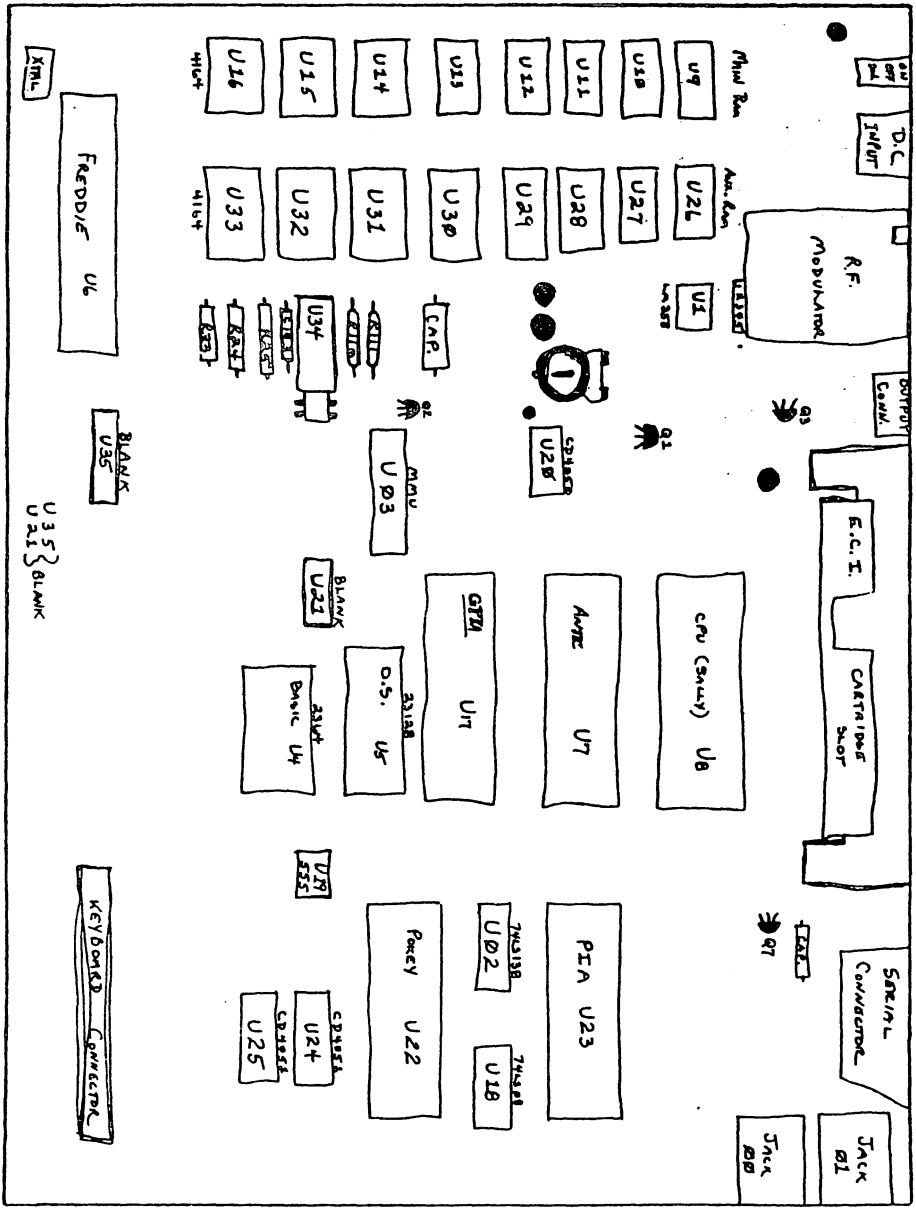
Program Listing
(produced with MegaFont II from XLENT)

```
10 REM ****************************
20 REM *   XEMEG RAMDISK INIT     *
30 REM * This program written for*
40 REM *    CL.A.U.G.  (C) 1986  *
50 REM *    (TM) Version 1.2     *
60 REM ****************************
70 DIM A$(6000),CIO$(40),RD$(10):C1=1:C2=C1+C1:C4=C2+C2:C5=C4+C1:C7=C5
+C2:C8=C4+C4:C9=C5+C4:C752=752:POKE 709,C9+C1
80 RESTORE 510:FOR I=C1 TO 40:READ A:CIO$(I,I)=CHR$(A):NEXT I:POKE 82,
C0:? :RD=ASC("k"):C10=C9+C1:POKE C752,C1
90 ? CHR$(125);:POKE 710,C0:? "MyDos version 4.2/XEMeg Initialization"
:? " ";:FOR I=C1 TO 38:? CHR$(13);:NEXT I:?
100 ? :? :? " Set-up for:":POKE 82,C9:? :? "1) 256K RAMdisk":? :? "2)
512K RAMdisk":POKE 702,C8*C8:POKE 694,C0
110 ? :? "3) 768K RAMdisk":? :? "4)1024K RAMdisk":POKE 82,C0:POKE C752
,C0
120 OPEN #C1,C4,C0,"K:":? :? :? "Selection (1-4)->";
130 GET #C1,I:IF I<C10*C4+C9 OR I>C10*C5+C2 THEN ? CHR$(253);:GOTO 130

140 A$=CHR$(I):? A$;:SEL=VAL(A$):RDSIZE=SEL*256:POSITION C0,C2+C1:POKE
 C752,C1:FOR I=C1 TO C10+C5:? CHR$(156);:NEXT I:?
150 POKE 752,C0:? :? :? "Are the ORIGINAL, extra 64K":? "RAM chips ins
talled (y/n)->";
160 GET #C1,I:IF I=C10*C8+C9 OR I=C10*C7+C8 THEN 180
170 ? CHR$(253);:GOTO 160
180 A$=CHR$(I):? A$:IF A$="Y" THEN RDSIZE=RDSIZE+C8*C8
190 IF RDSIZE=1088 THEN MAXFLG=C1:MAX=C1
200 ? :? "Drive number for":? "RAMdisk (2 to 9)->";
210 GET #C1,I:IF I<C10*C5 OR I>C10*C5+C7 THEN ? CHR$(253);:GOTO 210
220 A$=CHR$(I):? A$;:POKE 1802,VAL(A$):REM * This is the location of t
he RAMdisk number for MyDos.
230 ? :? :? "Will you be using":? "BASIC XE (Y/N)->";
240 GET #C1,I:IF I=C10*C8+C9 OR I=C10*C7+C8 THEN 260
250 ? CHR$(253);:GOTO 240
260 A$=CHR$(I):? A$:IF A$="Y" THEN XEFLAG=128:RD=RD+XEFLG:RDSIZE=RDSIZ
E-C8*C8:MAXFLG=C0
270 RD$=STR$(RDSIZE):RD$(LEN(RD$)+C1)=CHR$(RD):RD$(LEN(RD$)+C1)="RD(C)
"
280 IF LEN(RD$)<C10 THEN RD$(LEN(RD$)+C1)=" ":GOTO 280
290 RESTORE 550:IF XEFLG THEN FOR I=C1 TO C4:READ A:NEXT I
```

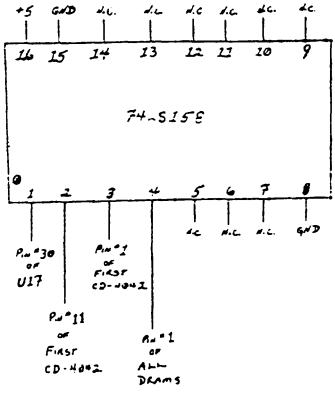
```
300 FOR I=C1 TO SEL*16:READ A:POKE 5511+I,A:NEXT I:IF MAXFLG THEN FOR
I=C1 TO C4:READ A:POKE 5575+I,A:NEXT I
310 IF MAXFLG THEN RESTORE 750:FOR I=5580 TO 5602:READ A:POKE I,A:NEXT
 I
320 IF MAXFLG THEN POKE 5642,193:GOTO 360
330 POKE 5595,C0:POKE 5596,215:REM *    Change control register addres
s.
340 POKE 5597,169:POKE 5598,128:REM *   Value to be 'OR'ed to turn Ban
king    OFF.
350 POKE 5602,255:REM *  Value to be    'OR'ed to turn Banking ON.
360 POKE 5640,RDSIZE/16:REM *  Size     of RAMdisk in 16K pages.
370 POKE 5658,C0:POKE 5659,215:REM *    Change Control Register Addres
s.
380 POKE 5607,C0:POKE 5608,215:REM *    Change Control Register Addres
s.
390 POKE C752,C1:POSITION C0,C4:FOR I=C1 TO C10:? CHR$(156);:NEXT I
400 ? :? :? "Now altering DUP.SYS, to reflect":? "changes in DOS.SYS."

410 OPEN #C2,C4,C0,"D:DUP.SYS":A=USR(ADR(CIO$),C2,ADR(A$),6000,C7)
420 IF PEEK(206)<>136 THEN POKE 82,C10:? :? CHR$(253);"CANNOT FIND DU
P.SYS":POKE 82,C0:POKE C752,C0:? :END
430 A$(A+C1)="":A$(280,289)=RD$:I=440:REM * Change top line of Mydos,
to      show the RAMdisk type.
440 TRAP I:CLOSE #C2:OPEN #C2,C8,C0,"D:DUP.SYS":A=USR(ADR(CIO$),C2,ADR
(A$),A,C10+C1):CLOSE #C2:GOTO 460
450 XIO 36,#C2,C0,C0,"D:DUP.SYS":I=40000:GOTO 440
460 IF MEFLG THEN ? :? A$(266,302):POSITION C10+C7+MAX,C10:? "^":? "Th
is indicates BASIC ME uses first 64K."
470 ? :? :? " Now go to DOS, and save your":? " altered version of Myd
os.":POKE C752,C0:? :? :END
480 REM * This is CIO$ in DATA form.
490 REM * It is a fast ML READ/WRITE
500 REM * for disk or cassette use.
510 DATA 104,104,104,10,10,10,10,170,104,157,69,3,104,157,68,3,104,157
,73,3,104,157,72,3
520 DATA 104,104,157,66,3,32,86,228,189,72,3,133,212,189,73,3,133,213,
189,67,3,133,206,96
530 REM * These are the RAMdisk pages
540 REM * in 16K blocks.
550 DATA 0,1,2,3
560 DATA 4,5,6,7
570 DATA 8,9,10,11
```

```
580 DATA 12,13,14,15
590 DATA 16,17,18,19
600 DATA 20,21,22,23
610 DATA 24,25,26,27
620 DATA 28,29,30,31
630 DATA 32,33,34,35
640 DATA 36,37,38,39
650 DATA 40,41,42,43
660 DATA 44,45,46,47
670 DATA 48,49,50,51
680 DATA 52,53,54,55
690 DATA 56,57,58,59
700 DATA 60,61,62,63
710 DATA 64,65,66,67
720 REM * This is the Modification to
730 REM * Mydos to get the extra 64K
740 REM in a 1088K system.
750 DATA 72,152,9,128,74,133,51,120,169,0,141,14,212,106,133,50,104,16
8,9,255,133,49,234
```

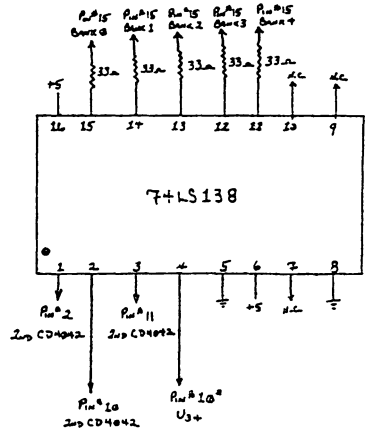
XEMeg Diagrams



REFRESH & 64K DRAM SELECT



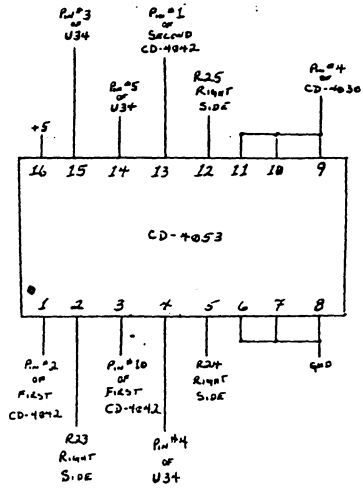
BANK SELECT



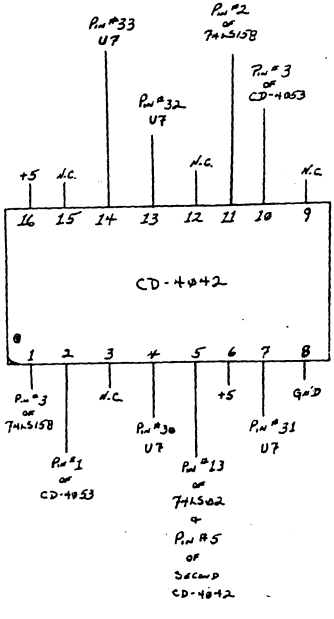
* ON U34, COUNT THE ACTUAL PINS. NOT THE HOLES.!!

NOTE: THE 2ND CD-4042 HAS D4 TO D7, (PINS 38, 39, 40 OR 1/2).

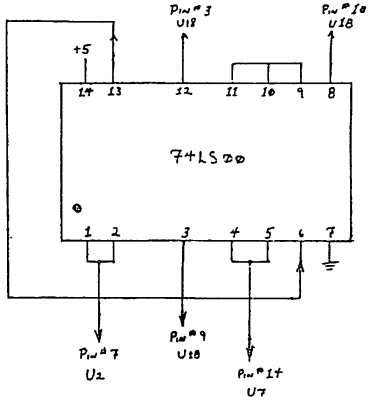
130 XE Compat. Switch



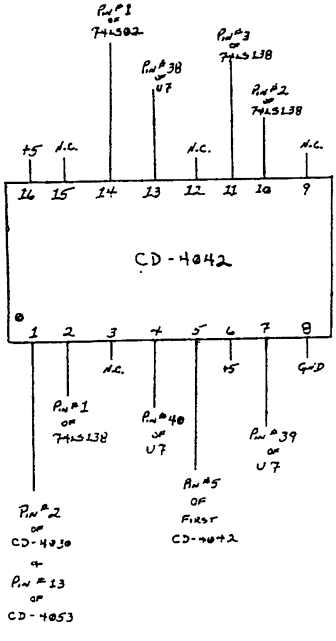
First Data Latch



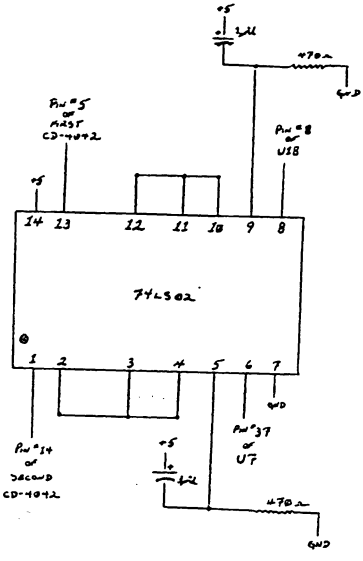
Clock Enable For Data Latch



Second Data Latch



POWER - UP INIT.

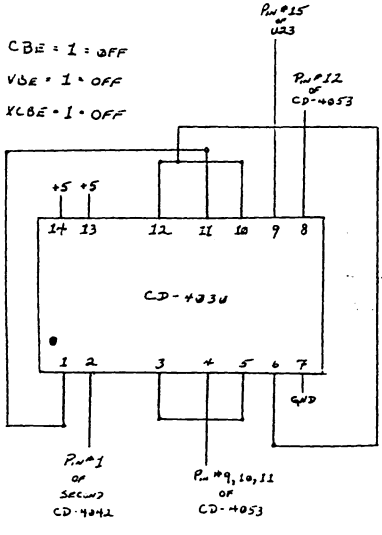


SWITCH CONTROL

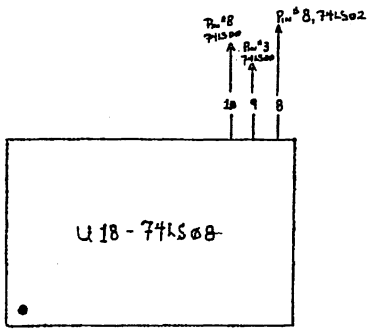
CBE = 1 = OFF

VBE = 1 = OFF

XCBE = 1 = OFF



CLOCK ENABLE (ATARI CHIP)



Newell 256K Upgrade Fix
by Wess Newell

This is a fix for the Newell 256K Upgrade for the 800XL for use with SpartaDos or MyDos.

Parts needed:
1. SPDT switch of your choice.
2. 12" jumper wire.

Procedure:
1. Remove mother board from case and remove shield.
2. Locate jumber installed on Newell board that runs to pin 15 of the PIA and remove it.
3. Run a new jumper from P85 to the common lug of the SPDT switch.
4. Run a new jumper from pin 15 of the PIA to one side of the SPDT switch.
5. Run a new jumper from the other side of the SPDT switch to pin 6 of U19.
6. Run a new jumper from pin 5 of U19 to pin 15 of the PIA.

Switched to one side, the RAM works like before. SpartaDos will only see what it thinks is a 130XE. Switch to the other side and use SpartaDoa "RD.COM" to use the full 256K of RAM.

This information was supplied by Wess Newell, Newell Industries, 214-442-6612.

Reprinted from the October 86 issue of Denver's MILE HIGH Atari Magazine.

XE Audio Fix
by Rich Andrews

There have been a few people that have noticed that the audio output of the 130XE is significantly lower than its sister 800XLs. As you might imagine, there is a fix. Locate R8, a 51K Ohm resistor. It is the second resistor to the right of the 1M358 audio chip. That chip is located in front of the video modulator. Tack solder a 0.47 microfarad capacitor across it. Now your levels should be on a par with the 800XL machines.

800XL Keyboard Repair

This article was condensed from a message on BitNet, downloaded by Red Varnum on April 8, 1986. It was written by someone named Jackson from Transaction Technology, Inc. (CitiCorp), Santa Monica, CA. Our thanks to him or her.

In many cases, bad keys on an 800XL are the results of poor internal spring adjustment. Readjusting keys is an inexpensive fix, so check it out before you rush out and buy a new keyboard.

The keys of an 800XL is actually a membrane pad with individual key contacts activated by springs depressed by the full travel keycap. The force applied by the spring must be enough to compress the contact pad, but not too much so the key doesn't feel too stiff. The springs are probably correct at the time of manufacture, but weaken with age. The fix is to "stretch" the spring a little, so that its rest-length is somewhat longer, resulting in more force when the key is depressed.

The keyboard membrane assembly seems to be well protected from internal contamination such as dust or dirt (but probably not liquid), although several years of normal aging might corrode the contacts. Problems with key spring tension are common, and likely if one or more isolated keys don't function. (However, if several adjacent keys in a row or column are all bad, then it might be the plug/jack, since the keys are sensed by a matrix of contacts.)

Assuming the key spring is at fault, the fix is roughly as follows:

0. DON'T disassemble the cabinet!

1. Power up the 800XL (with display, but without peripherals or cartridge), with the 'option' key depressed (or in BASIC, enter "bye")-- this should give the self-diagnostic menu.

2. Select the keyboard-test option, and manually test each key, noting which keys don't function.

3. Turn the power off.

4. Remove the keycap (see note below) and assembly from on of the bad keys. There are several parts (the keycap, the spring, and one or two more). If you have strong fingers, the keycap can be lifted off directly by holding it. Otherwise, try to gently pry it loose using some sort of tool hooked under the keycap edge, preferably with two hooks on opposite sides of the cap. An IC extractor will work. Don't squeeze the keycap very hard - it is soft plastic. The other parts may come out with the keycap, or you may need tweezers for them. If you look in the hole where the keycap was, you will see the membrane pad at the bottom.

5. Power up and enter the keyboard-test routine again.

6. Insert a Q-tip (or other soft but stiff probe) into the key hole, so it touches the plastic membrane pad. Press the Q-tip with moderate force - about the same pressure you would use to depress a normal key. The screen should indicate key contact. (If not, the problem is

elsewhere.)

7. Measure the length of the spring, and stretch it enough so that its new length at rest is about 1/8" longer (do it in small increments - springs like this are easy to permanently stretch, but very difficult to "unstretch").

8. Reassemble the keycap and parts - this can be done with power on - the key should now register correctly. If not, the spring may need more stretching, so repeat steps 4 - 8 on this key. (Also, the keycap and other parts might be sticking somewhat, and a very small amount of lubrication (e.g. vaseline) might help - but try to avoid this.)

9. Do steps 4 - 8 on each remaining non-functional key.

Steps 3 and 5 (power down and up) are not really necessary if you are careful - removing the keycaps should have no affect other than multiple key strokes. However, you might want to have power off the first time you try to remove a keycap.

NOTE: Most of the keycaps are plastic, but the 'RESET', 'OPTION', etc. keys are flimsy metal. These keys are easy to bend, but this can be avoided if the cabinet is disassembled first, which gives plenty of room to push the key tops out from below. It is better to reassemble the cabinet before powering up again. Opening the cabinet doesn't help with the plastic keys because they are all packed together, so don't go to the trouble of cabinet disassembly for them, unless you really need to.

XM301 Fix

by Jim Jenkins, downloaded from ATARI BASE BBS

This is a quick fix for XM301's that don't allow loading from the disk drives. It's accomplished by installing 470 ohm resistors on the following lines inside the XM301: 3, 9, 13.

Disassemble the case. Locate and unsolder the lines. (They're numbered.) Insert the resistor on the board, standing up, and solder from the bottom, clipping the excess. Take the unsoldered lines and connect them to the proper resistors on the top, close to the resistor. Clip the excess and re-assemble the case. It should work fine.

XE Console Key Fix

by Alan Haskell, by way of John Nagy of the CHAOS BBS (517) 371-1106

After having trouble with the console keys on the Atari 130XE used by the CHAOS BBS, and after being dissatisfied with the temporary nature of simply cleaning the contacts, I tried a few hardware fixes I ran across. Most didn't work, but one that did was published in the April/May issue of SPACE GAZETTE. The following is a slightly modified version of Alan Haskell's fix.

Items needed:

a small Phillips head screwdriver
a pair of needle nosed pliers
a small soldering iron
3 3000 ohm resistors, 1/4 watt
(The smaller the resistors are physically, the better they will fit.)

The Procedure

1. First making sure the power supply is disconnected, turn the XE over and remove the four screws that hold it together. Turn the machine back over and remove the top half of the case.
2. Lift out the keyboard and gently pull the ribbon out of its connector. Do NOT open the keyboard, as cleaning internally will not help, and you may damage the carbon tracks on the baking sheet inside.
3. remove the motherboard from the lower half of the case by removing the phillips screws holding it down.
4. Remove the top and bottom metal shields from the board by carefully straightening the bent tabs that go through the board.
5. Observe the top and underside of the connector you pulled the keyboard ribbon out of. As seen from the top as you would look at the computer normally, there are 24 connections, with connection #1 at the left and #24 at the right. Connection #3 (from the left) is the ground connection. The last 4, #21, #22, #23 and #24 are the START, SELECT, OPTION and RESET connections. When a connection is made from these points to the ground, the computer will know one or several of the keys are being pushed. Due to a design problem, the console keys have a bit too much resistance to always register even when you really lean on them. So, we will install the three resistors between the connections and ground so as to "leak" a bit more ground signal to the computer. That way, not nearly so good a connection is required by the actual keys when pressed.
6. Identify the proper pins on the bottom side of the circuit board. We will install the resistors under the board.
7. Solder one end of all three resistors together. Then connect this common end to the ground connection (pin #3). Then solder the free end of the resistors, one each, to pins 21, 22 and 23. (The RESET key never seems to be affected, nor any other keys.)
8. Be certain that the wires on the resistors do not touch each other nor any other circuitry! Use as little solder and as short a heating time as possible. Tape the resistors to prevent shorting if needed, and press them close to the circuit board.
9. Reassemble the shields to the board, bending back the tabs to hold it all together. Look inside tobe sure the resistors do not touch the lower shield.
10. Complete the re-assembly, taking the extra care not to stress or

insert the keyboard ribbon too many times. The spring contacts bend easily, and the conductive coating on the ribbon, if scratched through, will cause the keyboard to be useless. Don't worry too much, because that's difficult to cause if you are careful. (The original fix suggested pressing the the resistor leads into the connector so as to make it solder-free and simpler... but that "simplicity" ruined my connector and made hours more work! Solder it!)

11. Test the repair. Power up the machine and type in the following one lin BASIC program:

```
10 PRINT PEEK(53279):GOTO 10
```

Then type RUN. You will see a vertical row of 7's. Pushing OPTION should return 3's. SELECT will give you 5's and START will give you 6's. If each key responds every time, you have finished the repair.

This has permanently fixed the problem on dozens of XE's. It has restored my keys to "feather touch" after a long siege of having to boot 5 or 6 times mashing the OPTION key trying to get a boot without BASIC. It <u>will</u> work for you.

Hardwired Keypad

(This is a reprint from CompUTAH that they got from the Pocatello ACE group.)

NOTICE! This modification may void the warranty on your computer and keypad. You have been warned!

Several months ago, a friend found a great buy on a CX85 keypad for my Atari. After several weeks waiting for the keyboard handler software to load, I quit using the keypad altogether. Months went by and then one evening I tore my 800 apart and wired the keypad directly to my 800. Now I don't have to use the handler software at all. Whenever I need the keypad, I have it. It works with any software because the computer can't tell if the keypad is plugged in or not.

CAUTION! Do NOT attempt this modification if you do not feel comfortable with hardware modifications!!

To wire the keypad direct, you will need to open the CX85 and make sure it has switches. Some are made with a membrane pad and they will not work for this mod. If you have switches, de-solder them from the printed wire board and toss the board. To do this mod the neat way, you will need to buy a 15 pin male and female D jack and socket, some 13 conductor ribbon cable and a SPST switch.

Inside the 800, you will find that your keyboard is connected to the computer by a row of 16 pins. You will need to attach wires to 13 of these pins and connect them to the female 15 pin D jack as follows:

Pin 1 to D Pin 1	Pin 2 to D Pin 2
Pin 3 to D Pin 3	Pin 5 to D Pin 4
Pin 7 to D Pin 5	Pin 9 to D Pin 6
Pin 10 to D Pin 7	Pin 12 to D Pin 8
Pin 13 to D Pin 9	Pin 14 to D Pin 10
Pin 15 to D Pin 11	Pin 16 to D Pin 12
Pin 16 to D Pin 13	

With that done, close up your 800.

Now start on the keypad. Use the 13 conductor cable to wire to the other D jack, connecting one wire to each of the first 13 pins of the 15 on the jack. Then use an ohmeter to find the proper pins on the keypad to wire to the D jack. They should follow the list below.

D Pin to Keypad key pin listing

Pin One to one side of the keypad keys 0 7 8 9
Pin Two to one side of the keypad keys 1 2 3 4 5 6
Pin Three to one side of the keypad keys - =
Pin Four to one side of the keypad keys * +
Pin Five to one side of the keypad keys / . ,
Pin Six to one side of the keypad keys RET
Pin Seven to the other side of the keypad keys 7 6
Pin Eight to the other side of the keypad keys 8 RET
Pin Nine to the other side of the keypad keys 9 4 and one side of the SPST switch

Pin Ten to the other side of the keypad keys 3 0 and the other side of the SPST switch
Pin Eleven to the other side of the keypad keys / - + 2
Pin Twelve to the other side of the keypad keys = * 1 RET
Pin Thirteen to the other side of the keypad keys BACK
The center of the SPST switch to the other side of the keypad key . or ,

The following table may help. Each keypad key combines the specified computer pins (not the cable pins)

Key	D Pins	Key	D Pins
0	1 14	1	2 16
2	2 15	3	2 14
4	2 13	5	2 12
6	2 10	7	1 10
8	1 12	9	1 13
+	5 15	-	3 15
/	7 15	*	5 16
=	3 16	.	7 14
,	7 13	RET	12 16

For an explanation of the use of the SPST switch, read on. With the "." and "," keys, you should wire D pin 5. Then wire D pin 9 to one side of the SPST switch and D pin 10 to the other side of the SPST switch and wire the center to either the . key or the , key. This will allow you to use the key pad for decimal points or DATA statements, depending on which way the switch is thrown. Now mount the SPST switch in the back of the CX85 case and close up. Label which side of the switch is which and pat yourself on the back. The keypad keys should act just like the same keys on the keyboard. If they don't, go back and make sure all your wiring connections are correct.

4-Joystick XL/XE

by Mark J. Dulcey

When the Atari 1200XL appeared, one criticism was the new machine has only two joystick ports, rather than the four ports featured by the earlier 400 and 800 machines. This limitation is also present in all the newer Atari machines. Owners of these machines cannot play games intended for more than two players - until now!.

The interface described in this article allows you to connect four joysticks to your XL or XE machine. With the software included, it is compatible with most programs intended for the earlier computers. The interface plugs intothe joystick ports and requires no modification to your computer.

How the Interface Works

Lines 1 and 2 of joystick port 2 are programmed as outputs and used to select one of the four input jacks. The modified operating system outputs each of the four port addresses in turn. The data selectors in the interface connect the appropriate inputs to joystick port 1; the OS reads them from there. The stick values are put into the expected memory locations, so the interface is compatible with most software. Paddle inputs are connected directly from input ports 1 and 2 to the corresponding ports of the computer and are scanned normally.

Building the Interface

The schematic diagram and suggested board layout appear in Figure 1. The terminal points shown in the diagram are connected to pins of the connector plugs and jacks as indicated.

Parts List:

IC1	74LS21
IC2-4	74LS153
C1	22uf tantalum
C2-5	.1uf monolithic ceramic
C6-32	.001uf monolithic ceramic
R1-27	220 ohm, 1/4 watt
J1-4	DB9P
P1-2	DB9S

The Software

The software for the interface (Listing 1) modifies the Revision B operating system found on the Translator disk. It changes the joystick scanning code to work with the interface. It will work with either side of the Atari Translator disk, the Fix XL from ANTIC, or the Home-Made Translator from Analog. The Basic program creates an AUTORUN.SYS file. Put it on a disk with DOS. This is the 4-port handler disk.

If you have the Home-Made Translator, you can make a disk for use with the interface which boots in one step. Boot the HM Translator, then the handler disk. Load the BASIC program which creates the HM Translator and run it.

How to Use the Interface

To run programs with the 4-joystick interface, boot your translator (with BASIC enabled or disabled, as needed), then boot the 4-port

handler disk. Next insert your program disk and press (SELECT). Plug the interface into the joystick ports of your machine, and plug joysticks or other compatible controllers into the interface.

Limitations

Some programs will not work with this 4-joystick interface. Possible problems:

Reading the ports directly. This is usually done only for devices other than joysticks.

Changing any of the port lines to outputs. Devices which require this include keypad controllers (CX50s), MPP modems and printer interfaces.

Devices using the paddle inputs (paddles, CX85 keypad, touch tablet) will work only in the first two ports.

Any software which will not run with the OS in RAM. This is usually caused by protection schemes, not any inherent problem in the program.

Cartridge software is difficult to use with the Translator. Using a cartridge back-up program which converts them to disk files will let you use this driver.

The interface consumes about 20 ma. This should not be a problem unless you have substantial extra power supply loads on your machine.

If you have a RAMrod XL or similar board, you could burn the modified Translator into an EPROM and install it. This should solve the problems with protected software and cartridges.

The one piece of software I've not been able to get to work with this interface is M.U.L.E. There are at least two versions of the program. The earlier one appears to read the ports directly; the later one will not run with the OS in RAM. If anyone can tell me how to use this interface with M.U.L.E., I will be very grateful.

The schematic for this modification follows the program listing.

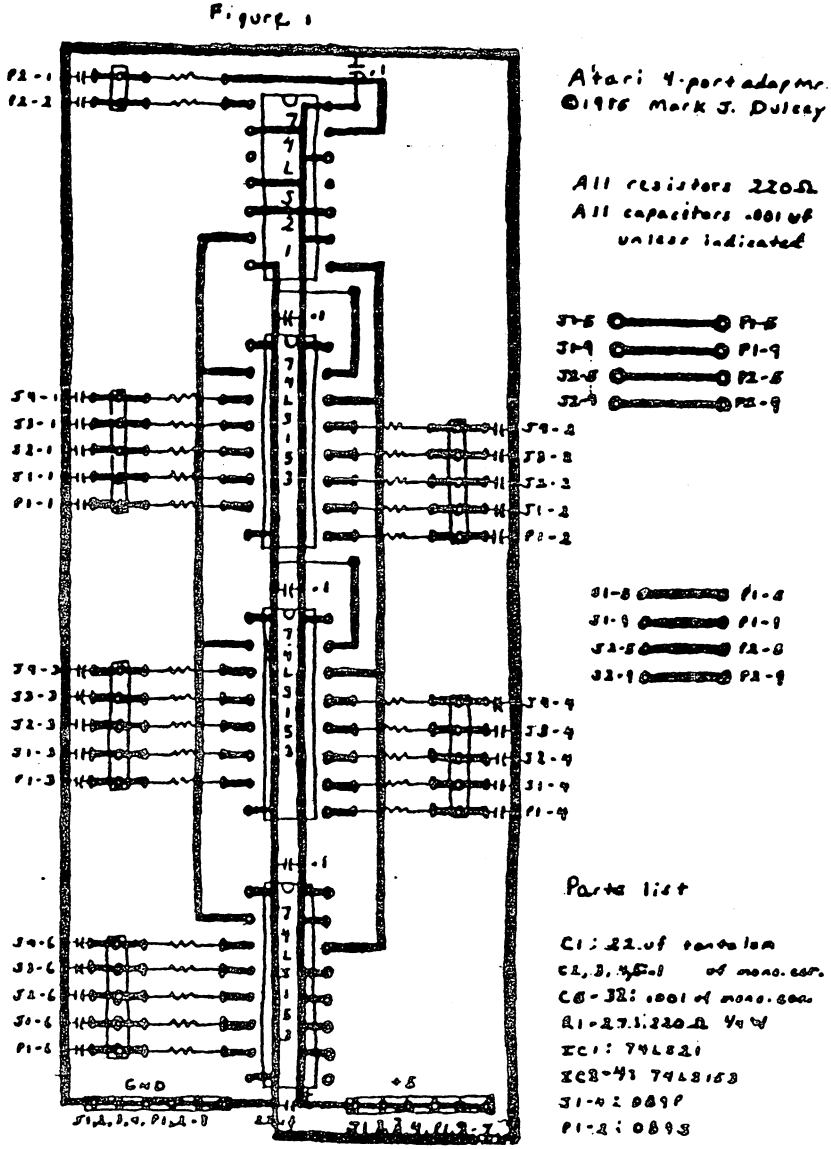
LISTING 1

(c) 1986 Mark J. Dulcey

```
10 OPEN #1,8,0,D:AUTORUN.SYS":TRAP 30
20 READ A:PUT #1,A:GOTO 20
30 CLOSE #1:END
1000 DATA 255,255,0,56,162,0,169,12
1010 DATA 157,66,3,32,86,228,162,0,169
1020 DATA 3,157,66,3,169,199,157,68,3
1030 DATA 169,56,157,69,3,169,8,157,74
1040 DATA 3,169,0,157,75,3,32,86,228
1050 DATA 162,0,169,11,157,66,3,169
1060 DATA 56,157,69,3,169,44,157,72
1070 DATA 3,169,0,157,73,3,32,86,228
1080 DATA 169,5,205,31,208,208,251,120
1090 DATA 169,0,141,14,212,141,14,210
1100 DATA 169,48,141,227,230,162,49
1110 DATA 189,105,56,157,115,232,202
1120 DATA 16,247,76,119,228,162,3,138
1130 DATA 10,10,10,10,141,0,211,234
1140 DATA 234,234,234,234,234,234,234
1150 DATA 234,234,234,173,0,211,41,15
1160 DATA 157,120,2,173,16,208,157,132
```

```
1170 DATA 2,189,0,210,157,112,2,189,4
1180 DATA 210,157,116,2,202,16,208,125
1190 DATA 155,70,111,117,114,45,112
1200 DATA 111,114,116,32,97,100,97,112
1210 DATA 116,101,114,32,100,114,105
1220 DATA 118,101,114,155,155,32,32,32
1230 DATA 80,114,101,115,115,32,83,69
1240 DATA 76,69,67,84,155,69,58,226,2
1250 DATA 227,2,0,56,224,2,225,2,0,56
```

Figure 1 -- Atari 4-port adapter
(c) 1986 Mark Dulcey



A-BUG

1050 Write Enable Switch

by Tom Kerschen

What this modification does is allows you to write to the back of a disk, or to any write protected disk, without notching the disk. The switch will allow you to set the normal operation of the drive, an ALWAYS write mode and a NEVER write mode.

Parts Needed

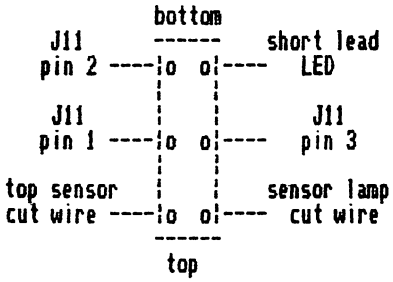
1 DPDT Switch with CENTER position OFF
(Radio Shack #275-620)
1 LED (Light Emitting Diode)
(Radio Shack #276-068)
6 feet of 24 guage insulated copper wire

Tools Needed

Small Needle-nose Pliers
Phillips Screwdriver
Low Wattage Soldering Iron
Small Drill
1/4 inch Drill Bit

Procedure

First, make sure the drive is disconnected from any power supply to keep yourself from frying. Remove the 6 screws from the bottom of the disk drive and take the top and front of the drive off. Now you need to decide where you would like to mount the switch. I have mine mounted in the front on the left-hand side with the LED right above it. These instructions will assume that position. You will need to position the switch and drill a 1/4 inch hole. Next, position the LED and drill a 1/4 inch hole for it, making sure to leave enough room for the switch on the back side. Solder a 16 inch piece of wire onto each of the 6 terminals of the switch. Wire in the switch and LED as per the following diagram:



To make the drawing make sense, the J11 connector is located in the back left of the main board of the drive. It is the 4 wire connector closest to the front. The wires are numbered 1 to 4 starting at the front of the drive. Unhook the connector. Cut the wire on connector pin 1. Solder the wire coming from the middle terminal of the switch to the connector. The other end of the wire you just cut solders to the wire that's labeled "top sensor cut wire" on the diagram. The wire from the switch labeled "J11 pin 2" piggyback solders on connector pin 2. Cut the wire on connector pin 3. Solder the wire from the other middle terminal of the switch to the now empty connector. The other end of the wire just cut is soldered to the wire labeled "sensor lamp cut wire" on the diagram. Piggyback solder a wire to connector pin 4 and connect it to the long lead on your LED. Connect the wire labeled "short lead LED" to the short lead on your LED.

That completes the wiring. Make sure all your connections are properly made and that the connector J11 is hooked back up. Reassemble the case, making sure the new wires don't get pinched under the sides or between the screw stands.

You no longer need to punch the notch into your disk to write to the back. With the switch down, you are in the NORMAL position. In the middle, you are in the NEVER position. And with the switch up, you are in the ALWAYS position. If you have any problems or questions, call the PALACE BBS (316) 683-4605 and leave mail to the sysop or SPARKY (co-sysop) and I'll try to answer them for you. Enjoy!

Shared Drives

by Roger Brandt

(This is reprinted from the March 1986 issue of SCAT.)

With the prices on 8-bit computers dropping, I have ended up with a number of computers and too few disk drives or printers to go around. So, I made a cable to connect two computers to one or two disk drives, so two people can play games simultaneously, or to access one disk with two different programs without reloading programs.

If you want to get more use out of your system, this is how I do it. I do want to warn you, though. I could not find the technical data to know for sure if this hookup is completely compatible, so I make no guarantees. But I have it hooked up and use it with a 130XE and an 800. This hookup has also worked successfully on an 800 and a 400, and on two 800s. The disk drives that I have used are an Atari, an Indus and a Rana, both alone and daisy-chained. So far, all systems work fine.

You will also have to have two monitors/tvs or an antenna switch to view each computer's output. Here's what each cable pin is used for, with a 13-pin plug:

1 - Clock In	2 - Clock Out
3 - Data In	4 - Ground
5 - Data Out	6 - Ground
7 - Command	8 - Motor Control
9 - Proceed	10 - +5 volts/Ready
11 - Audio In	12 - +12 volts
13 - Interrupt	

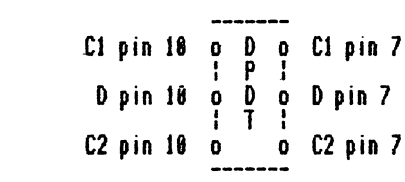
To make the new cable, use 2 pre-made cables, one 3 foot, the other 6 foot. Cut the 3 foot cable in half and cut 1.5 feet off the 6 foot cable, leaving 4.5 feet on the other end. Obtain a 2 foot length of 6 conductor cable and connect one end of each of the conductors to each terminal of a DPDT switch. Now mount the switch in a box which will sit in front of your disk drive. You now have one switch with cable attached and four plugs with cable attached (only 3 will be used; set aside one with a 1.5 foot cable length for spare parts). Mark the plug with the 4.5 foot length of cable with a "C" for computer and one with a 1.5 foot cable length with another "C" for the second computer and the last one with a 1.5 foot cable length with a "D" for disk drive.

Take the case off the plug to determine the color of the wires to each pin. (Or use an volt/ohm meter or continuity checker to determine wire color.) Make a chart for future reference. There is no standard system, so be careful. Find each of the 3 wires attached to pin 1 of each plug, splice them together and label them "1". Repeat the procedure with the wires for pins 2,3,4,5,6,9 and 13, labeling them as you go. Now, splice the wire from pin 12 of one of the computer plugs to pin 12 of the disk plug. Repeat with pin 11. (Only one 12 volt power supply should be used (pin 12) and one audio source (pin 11).)

Tape the wires from pins 11 & 12 of the other computer to prevent shorting. Splice the wire from pin 10 of one computer plug to the wire connected to one corner of the DPDT switch. Connect the pin 10 wire

from the disk drive to the center tap of the DPDT switch. Connect the remaining pin 10 wire from the other computer to the corner of the DPDT switch. Now, do the same thing with the wires from pin 7, using the other side of the DPDT switch.

The switch should look like this:



Now solder all the splices and tape over them to prevent shorting. Use a continuity checker to check your work. If everything is okay, wrap tape around all the splices to make it a neat package. Double check your work before you plug anything in to your computers and drive(s). That's it. You're done.

Wiring chart:

Computer 1 pin		Drive(s) pin		Computer 2 pin
1	--------	1	--------	1
2	--------	2	--------	2
3	--------	3	--------	3
4	--------	4	--------	4
5	--------	5	--------	5
6	--------	6	--------	6
7	---//---	7	---//---	7
8	--------	8	--------	8
9	--------	9	--------	9
10	---//---	10	---//---	10
11	-- --	11	--------	11
12	--------	12	-- --	12
13	--------	13	--------	13

-------- = wired ---//--- = switched
-- -- = not connected

1050 Double Density Fix

by Gordon Wantz

(Reprinted from the 01' Hackers Newsletter, November 86.)

If you have installed ICD's US Doublers in your 1050 drive and had such problems as loss of formatting and I/O errors, those problems are probably due to the drive overheating.

I have done some research and have found that the 6810 memory chip is greatly under-rated for the heat generated inside a 1050 drive. Double density seems to compound the problem. The simple and inexpensive fix for this is to remove the black box covering the double 6810 chip and attach a heat sink.

I used a heat sink purchased from Radio Shack (#276-1363) for 79 cents. I used 5-minute epoxy to attach the sink to the 6810 chip. There are heat transfer glues available, but the epoxy has worked adequately under the most severe testing.

No removal of the drive assembly is necessary if you're careful, but the 6810 chip must be removed so you can pry off the cover. BE CAREFUL REMOVING THE COVER!! After the sink is glued to the top of the double 6810 chip, simply re-install the chip in the socket in the exact same attitude is was originally. (Pin 1 towards the front.) Now replace the cover of the drive and you'll have a very reliable unit.

NOTE: There are high temperature versions of the PIA chip and the Microprocessor chip available from Rockwell Ind. I recommend these if your drive is getting extended use, such as a BBS, etc.

If you have any questions, call me at the Graveyard BBS (516) 783-7034 and I'll try my best to help you.

From the Editor

I'd like to thank all of the authors for doing these things, but most of all, for writing about them. Sharing how something was done says a lot about the person that did it, and they deserve our respect, admiration and thanks.

Also, thanks to those of you who contributed articles from whatever sources they were found. I tried to site the source when I knew it. If something looks familiar, and you didn't get credit, I apologize. Let me know who you are, and I'll fix it in the next printing.

And thanks to all the Atari owners who refuse to accept the limits placed on us by the machine maker. We are unique in our desire to push the limits, it is commendable that we do so and we shouldn't let anyone tell us otherwise. (Even if they are the ones who make the #@%& things.)

DOLLAR ATARI RULZ!

Gordie Meyer, editor
P.O. Box 1982
Ames, IA 50010-1982

Notes

More Notes

Midwest Atari Group- Iowa Chapter
Second Printing.- December 1987